```python
def water_left(astronauts, water_left, days_left):
    for argument in [astronauts, water_left, days_left]:
        try:
            # If argument is an int, the following operation will work
            argument / 10
        except TypeError:
            # TypError will be raised only if it isn't the right type
            # Raise the same exception but with a better error message
            raise TypeError(f"All arguments must be of type int, but received: '{argument}'")
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    if total_water_left < 0:
        raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days!")
    return f"Total water left after {days_left} days is: {total_water_left} liters"
```

✓ 0.8s

```python
water_left("3", "200", None)
```

⊗ 0.1s

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Untitled-1.ipynb Cell 13' in water_left(astronauts, water_left, days_left)
      3 try:
      4     # If argument is an int, the following operation will work
----> 5     argument / 10
      6 except TypeError:
      7     # TypError will be raised only if it isn't the right type
      8     # Raise the same exception but with a better error message

TypeError: unsupported operand type(s) for /: 'str' and 'int'

During handling of the above exception, another exception occurred:

TypeError                                 Traceback (most recent call last)
Untitled-1.ipynb Cell 14' in <module>
----> 1 water_left("3", "200", None)

Untitled-1.ipynb Cell 13' in water_left(astronauts, water_left, days_left)
      5         argument / 10
```

open.py

C: > Users > Jose > Desktop > open.py > ...

```python
def main():
    open("/path/to/mars.jpg")

if __name__ == '__main__':
    main()
```

```
C:\Users\Jose\Desktop>python3 open.py
Traceback (most recent call last):
  File "C:\Users\Jose\Desktop\open.py", line 5, in <module>
    main()
  File "C:\Users\Jose\Desktop\open.py", line 2, in main
    open("/path/to/mars.jpg")
FileNotFoundError: [Errno 2] No such file or directory: '/path/to/mars.jpg'

C:\Users\Jose\Desktop>
```

C: > Users > Jose > Desktop > config.py > main

```python
def main():
    try:
        configuration = open('config.txt')
    except FileNotFoundError:
        print("Couldn't find the config.txt file!")


if __name__ == '__main__':
    main()
```

```
C:\Users\Jose\Desktop>python config.py
Couldn't find the config.txt file!
```

```
try:
...     open('config.txt')
... except FileNotFoundError:
...     print("Couldn't find the config.txt file!")
```

✓ 0.5s

```
Couldn't find the config.txt file!
```

```
try:
...     open("mars.jpg")
... except FileNotFoundError as err:
...     print("got a problem trying to read the file:", err)
...
got a problem trying to read the file: [Errno 2] No such file or directory: 'mars.jpg'
```

⊗ 0.6s

```
  Input In [5]
    got a problem trying to read the file: [Errno 2] No such file or directory: 'mars.jpg'
        ^
SyntaxError: invalid syntax
```

```
try:
...     open("config.txt")
... except OSError as err:
...     if err.errno == 2:
...         print("Couldn't find the config.txt file!")
...     elif err.errno == 13:
...         print("Found config.txt but couldn't read it")
```

✓ 0.7s

```
Couldn't find the config.txt file!
```

```python
def water_left(astronauts, water_left, days_left):
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    return f"Total water left after {days_left} days is: {total_water_left} liters"
```

✓ 0.6s

```python
water_left(5, 100, 2)
```

✓ 0.9s

```
'Total water left after 2 days is: -10 liters'
```

```python
def water_left(astronauts, water_left, days_left):
    daily_usage = astronauts * 11
    total_usage = daily_usage * days_left
    total_water_left = water_left - total_usage
    if total_water_left < 0:
        raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} days
    return f"Total water left after {days_left} days is: {total_water_left} liters"
```

✓ 0.8s

```python
water_left(5, 100, 2)
```

⊗ 0.1s

```
---------------------------------------------------------------------------
RuntimeError                              Traceback (most recent call last)
Untitled-1.ipynb Cell 10' in <module>
----> 1 water_left(5, 100, 2)

Untitled-1.ipynb Cell 9' in water_left(astronauts, water_left, days_left)
      4 total_water_left = water_left - total_usage
      5 if total_water_left < 0:
----> 6     raise RuntimeError(f"There is not enough water for {astronauts} astronauts after {days_left} day
      7 return f"Total water left after {days_left} days is: {total_water_left} liters"

RuntimeError: There is not enough water for 5 astronauts after 2 days!
```